Tools for Undecidability

1. Diagonalization

   Halt is RE, Non-Recursive (Undecidable)

   Total is Non-RE (Not Semi-Decidable)

2. Reduction

   If $A \le_m B$ then

   A undec. $\Rightarrow$ B undec.

   A Non-RE $\Rightarrow$ B Non-RE

3. Rice's Theorem

   A property of program I/O behavior

   & A Non-trivial

   $\Rightarrow$ A Not-Recursive

4. Use of Quantification to set upper bound on A's complexity

   (Here complexity metrics are

   Solvable (Decidable, Recursive)

   RE, Non-Recursive

   Co-RE, Non-Recursive

   Non-RE, Non-Co-RE

RICE'S THEOREM (S)

THERE ARE THREE (3) CONSTRAINTS

1. IT IS ABOUT PROPERTIES OF COMPUTABLE FUNCTIONS (PROGRAMS, PROCEDURES) IT SPLITS THE INDICES OF FUNCTIONS INTO TWO CLASSES. FOR SOME PROPERTY, P, WE HAVE SETS S_P AND $\overline{S}_P$
   $S_P = \{ f \mid \varphi_f$ HAS PROPERTY $P \}$
2. IT CAN BE APPLIED ONLY TO NON-TRIVIAL PROPERTIES. P IS NON-TRIVIAL IF
   $S_P \neq \emptyset$ AND $\overline{S}_P \neq \emptyset$
3. IT MUST BE A PROPERTY ABOUT FUNCTIONAL BEHAVIOR, NOT IMPLEMENTATION OR EFFICIENCY. THAT IS, IF f AND g HAVE SAME BEHAVIOR THEN BOTH ARE IN S_P OR BOTH ARE IN $\overline{S}_P$

RICE'S VARIANTS BASED ON ~~BEAHAVIOR~~ TYPES

I. WEAK BEHAVIOR TYPE BASED ON DOMAIN

f AND g HAVE SAME BEHAVIOR IF
$Dom(f) = Dom(g)$ – STANDARD RICE

II. WEAK BEHAVIOR TYPE BASED ON RANGE

f AND g HAVE SAME BEHAVIOR IF
$Rng(f) = Rng(g)$

III. STRONG BEHAVIOR TYPE BASED ON ACTUAL MAPPINGS FROM INPUT TO OUTPUT

f AN g HAVE SAME BEHAVIOR IF
$\forall x \; f(x) = g(x)$
THIS MEANS SAME VALUE IF CONVERGE AND BOTH DIVERGE OTHERWISE

$\mathrm{III} \Rightarrow \mathrm{I}$ AND II
BUT $\mathrm{I} \not\Rightarrow \mathrm{III}$, $\mathrm{II} \not\Rightarrow \mathrm{III}$ AND
$\mathrm{I} \,\&\, \mathrm{II} \not\Rightarrow \mathrm{III}$

RICE'S THEOREM STATEMENT

IF P IS A NON-TRIVIAL PROPERTY OF FUNCTION INDICES THAT IS IMMUNE TO IMPLEMENTATION (AN I/O PROPERTY) THEN P IS UNDECIDABLE (NON-RECURSIVE)

KEYS TO PROOF ARE:

1. <u>ALL</u> FUNCTIONS WITH EMPTY DOMAINS/RANGES WILL EITHER HAVE PROPERTY P OR NOT HAVE PROPERTY P.
2. ASSUME ALL FUNCTIONS WITH EMPTY DOMAINS/RANGES ARE IN $\overline{S_P}$. IF NOT WE PROVE THE COMPLEMENT OF P TO BE UNSOLVABLE
3. AS P IS NON-TRIVIAL, THERE IS SOME FUNCTION INDEX IN S_P. CALL THIS INDEX r

PROOF OF RICE'S THEOREM

LET P MEET ALL CONDITIONS OF RICE'S THEOREM AND LET $r \in S_P$. WE KNOW THAT $\emptyset \in \overline{S_P}$ WHERE $\emptyset$ INDEX IS OF SOME FUNCTION THAT DIVERGES EVERYWHERE

CONSIDER $HALT = \{\langle x, y\rangle \mid \varphi_x(y)\downarrow\}$

DEFINE $f_{r,x,y}$

r AS ABOVE
x, y ARBITRARY

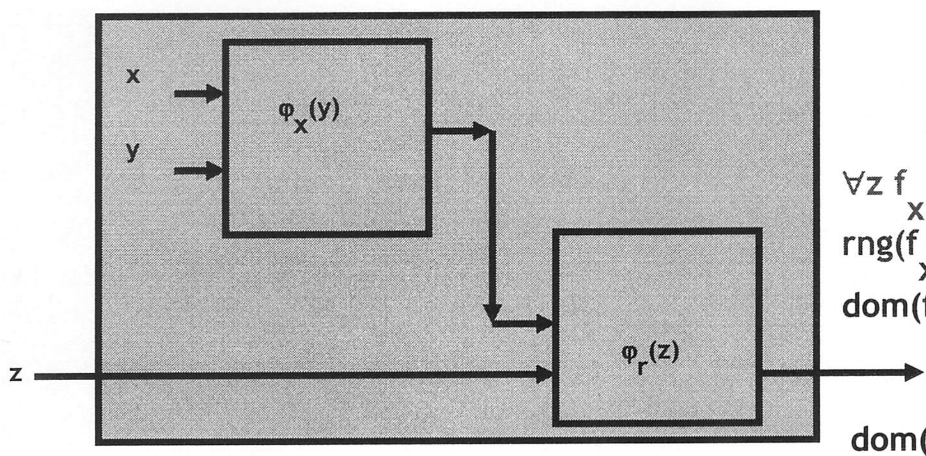
AS
$$f_{r,x,y}(z) = \varphi_x(y) - \varphi_x(y) + \varphi_r(z)$$

IF $\langle x,y\rangle \in HALT$ THEN
$f_{r,x,y}(z) = \varphi_r(z)$ AND $f_{r,x,y} \in S_P$

IF $\langle x,y\rangle \notin HALT$ THEN
$\forall z\, f_{r,x,y}(z)\uparrow$ AND $f_{r,x,y} \in \overline{S_P}$

THUS,
$$HALT \leq S_P$$
AND SO P IS NON-RECURSIVE

Rice's Picture Proof



$\forall z\ f_{x,y,r}(z)=\varphi_r(z)$ If $\varphi_x(y)\downarrow$ (RED)
$rng(f_{x,y,r})=rng(\varphi_r)$ If $\varphi_x(y)\downarrow$ (BLUE)
$dom(f_{x,y,r})=dom(\varphi_r)$ If $\varphi_x(y)\downarrow$ (BLACK)

$dom(f_{x,y,r})=\phi$ If $\varphi_x(y)\uparrow$ (BLACK
$rng(f_{x,y,r})=\phi$ If $\varphi_x(y)\uparrow$ (BLUE)
$\exists z\ f_{x,y,r}(z)\neq\varphi_r(z)$ If $\varphi_x(y)\uparrow$ (RED)

Black is for standard Rice's Theorem;
Black and Red are needed for Strong Version
Blue is just another version based on range

NOTE: IN ONLINE NOTES I USED
$f_{x,y,r}$
IN HANDWRITTEN NOTE I USED
$f_{r,x,y}$

Corollaries to Rice's

Corollary: The following properties of re sets are undecidable

a) L = Ø ($L \neq \phi$)

b) L is finite

c) L is a regular set

d) L is a context-free set

DO COMPLEMENTS IN ALL CASES

RICE EXAMPLES (WEAK)

$\text{HASZERO} = \{f \mid \exists x\ f(x)=0\}$

$C_0 \in \text{HASZERO}$; $C_1 \notin \text{HASZERO}$

SO NON-TRIVIAL

CAN USE VERSION II OF RICE

LET f AND g BE ARB. INDICES SUCH THAT

$\text{RNG}(f) = \text{RNG}(g)$

$f \in \text{HASZERO}$ IFF $\exists x\ f(x)=0$
IFF $0 \in \text{RNG}(f)$
IFF $0 \in \text{RNG}(g)$
IFF $\exists x\ g(x)=0$
IFF $g \in \text{HASZERO}$

BY RICE'S HASZERO IS NON-RECURSIVE

$\text{TOTAL} = \{f \mid \forall x\ f(x)\downarrow\}$

$C_0 \in \text{TOTAL}$ $\quad \mu y[y=y+1] \notin \text{TOTAL}$

SO NON-TRIVIAL

CAN USE VERSION I OF RICE

LET f AND g BE ARB. INDICES SUCH THAT

$\text{Dom}(f) = \text{Dom}(g)$

$f \in \text{TOTAL} \Leftrightarrow \forall x\ f(x)\downarrow \Leftrightarrow \text{Dom}(f) = \mathbb{N}$
$\Leftrightarrow \text{Dom}(g) = \mathbb{N} \Leftrightarrow \forall x\ g(x)\downarrow \Leftrightarrow g \in \text{TOTAL}$

BY RICE'S TOTAL IS NON-RECURSIVE

RICE EXAMPLE (STRONG)

$\text{MonoIncr} = \{f \mid \forall x\ f(x+1) > f(x)\}$

$S(x) = x+1 \in \text{MonoIncr}$

$C_0(x) = 0 \notin \text{MonoIncr}$

SO NON-TRIVIAL

MUST USE VERSION 3 OF RICE

LET f AND g BE ARB. INDICES SUCH THAT

$\forall x\ f(x) = g(x)$

$f \in \text{MonoIncr}$ IFF $\forall x\ f(x+1) > f(x)$

IFF $\forall x\ g(x+1) > g(x)$

AS $\forall x\ g(x) = f(x)$

IFF $g \in \text{MonoIncr}$

COMMENTS ABOUT UNIVERSAL MACHINE

$\text{CONFIG}(F, X, 0) = X$ (STARTING ID_0)

$\text{CONFIG}(F, X, y+1) = \text{NEXT}(F, \text{CONFIG}(F, X, y))$

(ID_{y+1} FOLLOWING ID_y)

$\text{STP}(F, X, t) = \text{CONFIG}(F, X, t) == \text{CONFIG}(F, X, t+1)$

WE HAVE CONVERGED AT t OR EARLIER

$\text{VALUE}(F, X, t) = \text{RESULT}(\text{CONFIG}, F, X, t)$

IF $\text{STP}(F, X, t)$

$\text{VALUE}(F, X, t) = \underline{\text{SOME ARBITRARY VALUE}}$

IF $\overline{\text{STP}(F, X, t)}$

TYPICALLY WE EVALUATE AS 0 IF VALUE IS MEANINGLESS

USEFUL FUNCTIONS (STP, VALUE)

WHEN DEVELOPING UNIVERSAL FUNCTION WE CREATED A PREDICATE STP TO SEE IF FUNCTION HALTS IN SOME FIXED TIME FOR SOME FIXED INPUT

$STP(f, x, t) = \text{TRUE}$

IFF $\varphi_f(x)$ CONVERGES IN t OR FEWER STEPS

WE ALSO NEEDED A WAY TO EXTRACT OUTPUT IF $\varphi_f(x)$ HALTS

$VALUE(f, x, t) = f(x)$

PROVIDED $STP(f, x, t)$

ELSE $VALUE(f, x, t)$ GIVES A MEANINGLESS VALUE (WE USE 0)

USES OF STP AND VALUE

$\langle f, x \rangle \in HALT \Leftrightarrow \exists t\, [STP(f,x,t)]$

$f \in TOTAL \Leftrightarrow \forall x\, \exists t\, [STP(f,x,t)]$

$f \in HasZero \Leftrightarrow \exists \langle x,t \rangle [STP(f,x,t)\ \&\&$
$VALUE(f,x,t) == 0]$

$f \in MonoIncr \Leftrightarrow \forall x\, \exists t\, [STP(f,x,t)\ \&\&$
$STP(f,x+1,t)\ \&\&$
$VALUE(f,x,t) <$
$VALUE(f,x+1,t)]$

$f \in EMPTY \Leftrightarrow \forall \langle x,t \rangle [\,\overline{STP(f,x,t)}\,]$

CATEGORIES (P ALGORITHMIC PREDICATE)

$x \in S \Leftrightarrow P(x)$ RECURSIVE

$x \in S \Leftrightarrow \exists y\, P(x,y)$ RE

$x \in S \Leftrightarrow \forall y\, P(x,y)$ CO-RE

$x \in S \Leftrightarrow \forall y\, \exists z\, P(x,y,z)$ NON-RE, NON-CORE

$x \in S \Leftrightarrow \exists y\, \forall z\, P(x,y,z)$ NON-RE NON-CORE

RE $\equiv$ SEMI DEC. (PART 1)

LET S BE RE THEN EITHER

$S = \emptyset$ OR $S = \mathrm{RNG}(f_S)$

WHERE f_S IS AN ALGORITHM

WE CAN BUILD g_S AS

$$g_S(x) = \mu y\,[y = y+1] \text{ IF } S = \emptyset$$

OR AS

$$g_S(x) = \mu y\,[f_S(y) = x]$$

OR

$$g_S(x) = \exists y\,[f_S(y) = x]$$

OR

$$g_S(x) = (\exists y\ f_S(y) = x) * x$$

IN ALL THREE CASES

$$\mathrm{Dom}(g_S) = S$$

IN THIRD CASE

$$\mathrm{Dom}(g_S) = \mathrm{RNG}(g_S) = S$$

THUS,

RE $\Rightarrow$ SEMI DEC.

RE ≡ SEMI-DEC. (PART 2)

LET S BE SEMI-DECIDABLE THEN

$S = \mathrm{Dom}(g_S)$

WHERE g_S IS A PROCEDURE

WE CAN BUILD f_S IF WE ASSUME

$S \neq \emptyset$

WE CAN DO SO SINCE IF $S = \emptyset$ THEN IT IS RE BY DEFINITION

LET $a \in S$ BE AN ARB. ELEMENT OF S

$$f_S(\langle x, t \rangle) = \mathrm{STP}(g_S, x, t) * x + (1 - \mathrm{STP}(g_S, x, t)) * a$$

f_S IS AN ALGORITHM

$\mathrm{RNG}(f_S) = \mathrm{Dom}(g_S)$

NOTE: f_S ENUMERATES EACH ELEMENT OF S INFINITELY OFTEN

RESULT IS

SEMI-DEC $\Rightarrow$ RE

NOTE: f_S IS <u>PRIMITIVE</u> REC.

REC & RE

LET S BE A RECURSIVE (DECIDABLE) SET

S HAS AN ASSOCIATED ALGORITHMIC PREDICATE

χ_S, CALLED ITS CHARACTERISTIC FUNCTION SUCH THAT

$$x \in S \iff \chi_S(x)$$

THE COMPLEMENT OF S, $\bar{S}$, IS ALSO RECURSIVE AND

HAS IT'S OWN CHARACTERISTIC FUNCTION, $\chi_{\bar{S}}$

$$x \in \bar{S} \iff \chi_{\bar{S}}(x) \iff \chi_S(x) = 0$$

S REC $\Rightarrow$ S IS RE && $\bar{S}$ IS RE

CAN DEFINE g_S & $g_{\bar{S}}$ WHERE $S = \mathrm{Dom}(g_S)$, $\bar{S} = \mathrm{Dom}(g_{\bar{S}})$

BY

$g_S(x) = \mu y\, \chi_S(x)$ - DIVERGE IFF $\overline{\chi_S(x)}$

$g_{\bar{S}}(x) = \mu y\, \chi_{\bar{S}}(x)$ - DIVERGE IFF $\overline{\chi_{\bar{S}}(x)}$ IFF $\chi_S(x)$

S REC $\iff$ S & $\bar{S}$ ARE BOTH RE

ON PREVIOUS PAGE WE SHOWED

S REC $\Rightarrow$ S & $\bar{S}$ ARE BOTH RE

NEED

S RE & $\bar{S}$ RE $\Rightarrow$ S REC

S RE $\iff$ (i) $S = \emptyset$ OR (ii) $S = RNG(f_S)$, f_S ALG.

ALSO S RE $\iff$ $S = Dom(g_S)$, g_S A PROCEDURE

WE WILL USE $S = Dom(g_S)$, g_S A PROC.

AS $\bar{S}$ IS RE $\bar{S} = Dom(g_{\bar{S}})$, $g_{\bar{S}}$ A PROC.

$$\chi_S(x) = STP(g_S, x, \mu t[STP(g_S, x, t) \| STP(g_{\bar{S}}, x, t)])$$

COULD DO OTHER WAY

IF $S = \emptyset$ THEN $\forall x\, \chi_S(x) = 0$ $\Big\}$ REC

IF $S = \mathbb{N}$ THEN $\forall x\, \chi_S(x) = 1$

ASSUME $S \neq \emptyset$ AND $S \neq \mathbb{N}$ ($\bar{S} \neq \emptyset$)

AND $S = RNG(f_S)$ $\bar{S} = RNG(f_{\bar{S}})$

THEN

$$\chi_S(x) = f_S(\mu y[f_S(y) == x \,\|\, f_{\bar{S}}(y) == x]) == x$$

re Characterizations

Theorem: Suppose **S** $\neq \emptyset$ then the following are equivalent:

1. **S** is re
2. **S** is the range of a primitive rec. function
3. **S** is the range of a total recursive function
4. **S** is the domain of a partial rec. function
5. **S** is the range/domain of a partial rec. function whose domain is the same as its range and which acts as an identity when it converges

Quantification #1

- **S** is decidable iff there exists an algorithm χ_S (called **S**' s characteristic function) such that
  $\mathbf{x \in S \Leftrightarrow \chi_S(x)}$
  This is just the definition of decidable.

- **S** is re iff there exists an algorithm $\mathbf{A_S}$ where
  $\mathbf{x \in S \Leftrightarrow \exists t\ A_S(x,t)}$
  This is clear since, if $\mathbf{g_S}$ is the index of a procedure that semi-decides **S,** then
  $\mathbf{x \in S \Leftrightarrow \exists t\ STP(g_S, x, t)}$
  So, $\mathbf{A_S(x,t) = STP_{g_S}(x, t)}$, where $\mathbf{STP_{g_S}}$ is the **STP** function with its first argument fixed.

Quantification#2

- **S** is re iff there exists an algorithm $\mathbf{A_S}$ such that
  $\mathbf{x \notin S \Leftrightarrow \forall t\ A_S(x,t)}$
  This is clear since, if g_S is the index of the procedure ψ_S that semi-decides **S**, then
  $\mathbf{x \notin S \Leftrightarrow \sim\exists t\ STP(g_S, x, t) \Leftrightarrow \forall t\ \sim STP(g_S, x, t)}$
  So, $\mathbf{A_S(x,t) = \sim STP_{g_S}(x, t)}$, where $\mathbf{STP_{g_S}}$ is the **STP** function with its first argument fixed.
- Note that this works even if **S** is recursive (decidable). The important thing there is that if **S** is recursive then it may be viewed in two normal forms, one with existential quantification and the other with universal quantification.
- The complement of an re set is **co-re**. A set is recursive (decidable) iff it is both re and co-re.

Quantification#3

- The **Uniform Halting Problem** was already shown to be non-re. It turns out its complement is also not re. In fact, we can (but won't) show that **TOTAL** requires an alternation of quantifiers. Specifically,

  $f \in \text{TOTAL} \Leftrightarrow \forall x \exists t\ (\ \text{STP}(\ f, x, t\ )\ )$
  and this is the minimum quantification we can use, given that the quantified predicate is recursive.

USES OF STP AND VALUE

Intentionally Repeated to Emphasize Quantification

$$\langle f, x\rangle \in HALT \Leftrightarrow \exists t\,[STP(f,x,t)]$$

$$f \in TOTAL \Leftrightarrow \forall x\, \exists t\,[STP(f,x,t)]$$

$$f \in HasZero \Leftrightarrow \exists \langle x,t\rangle\,[STP(f,x,t)\ \&\&\ VALUE(f,x,t)==0]$$

$$f \in MonoIncr \Leftrightarrow \forall x\, \exists t\,[STP(f,x,t)\ \&\&\ STP(f,x+1,t)\ \&\&\ VALUE(f,x,t) < VALUE(f,x+1,t)]$$

$$f \in EMPTY \Leftrightarrow \forall \langle x,t\rangle\,[\,\overline{STP(f,x,t)}\,]$$

CATEGORIES (P ALGORITHMIC PREDICATE)

$x \in S \Leftrightarrow P(x)$ RECURSIVE

$x \in S \Leftrightarrow \exists y\, P(x,y)$ RE

$x \in S \Leftrightarrow \forall y\, P(x,y)$ CO-RE

$x \in S \Leftrightarrow \forall y \exists z\, P(x,y,z)$ NON-RE, NON-CORE

$x \in S \Leftrightarrow \exists y \forall z\, P(x,y,z)$ NON-RE NON-CORE

REAL-TIME (CONSTANT EXEC. TIME)

$RT = \{ f \mid \exists C \ni \forall x\, f(x)\downarrow$ IN AT MOST C STEPS $\}$

$f \in RT \Leftrightarrow \exists C\, \forall x\, STP(f, x, C)$

CANNOT USE RICE'S THEOREM TO SHOW RT UNDEC. WHY?

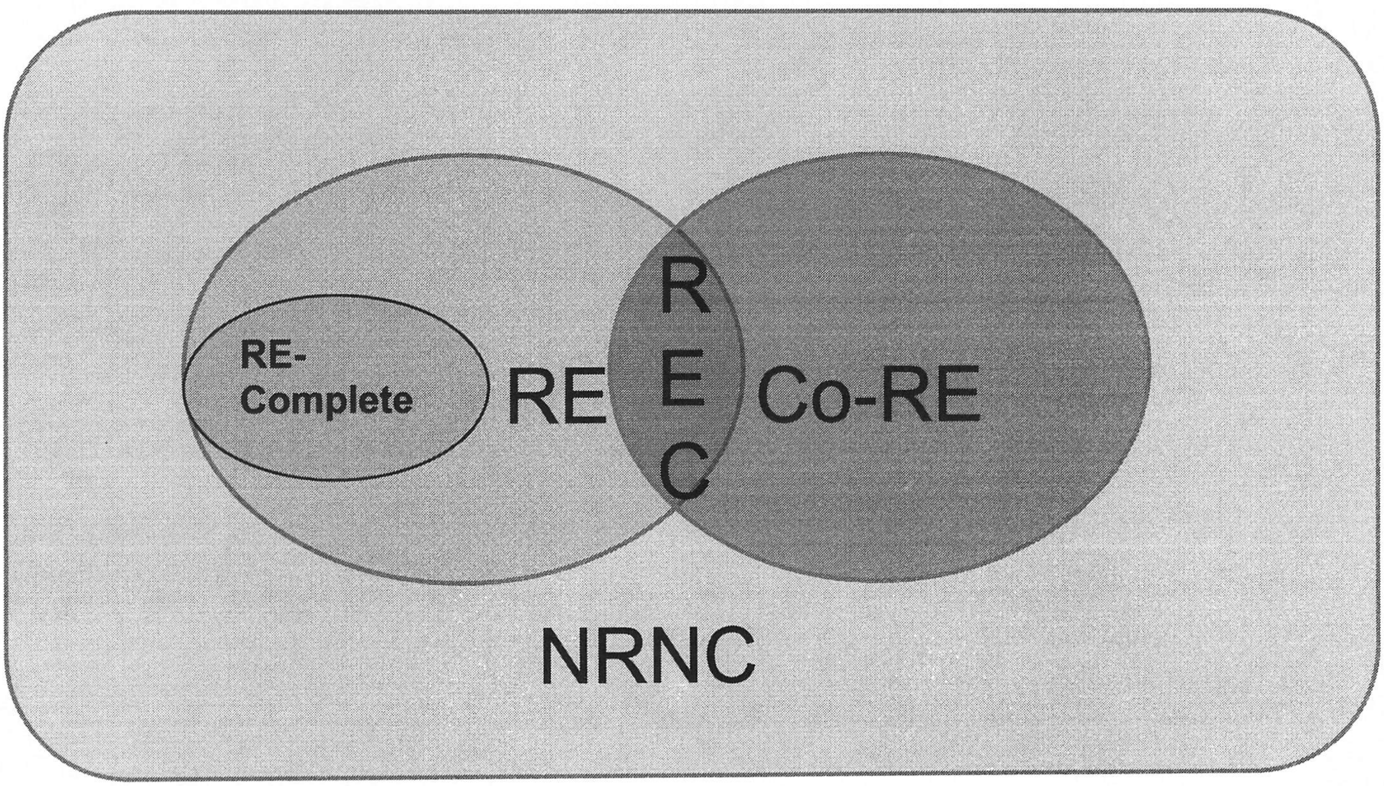
IS RT RE?

$RT = \{ f \mid \exists C \ni \forall x\ f(x)\downarrow$ IN AT MOST C STEPS $\}$

$\exists C \forall x_{|x| \le C}\ f(x)\downarrow$ IN C STEPS

From above we see that
Constant Execution Time is RE
(semi-decidable)

UNIVERSE OF SETS



NonRE = (NRNC ∪ Co-RE) - REC

Sample Question#1

1. Given that the predicate **STP** and the function **VALUE** are algorithms, show that we can semi-decide

   HZ = { f | φ_f evaluates to 0 for some input}

   Note: **STP(f, x, s)** is true iff φ_f**(x)** converges in **s** or fewer steps and, if so, **VALUE(f, x, s) = φ_f(x)**.

Sample Questions#2,3

2. Use Rice's Theorem to show that **HZ** is undecidable, where **HZ** is

   HZ = { f | φ_f evaluates to 0 for some input}

3. Redo using Reduction from **HALT**.

Sample Question#4

4. Let **P = { f | ∃ x [STP(f, x, x)] }**. Why does Rice's theorem not tell us anything about the undecidability of **P**?

Sample Question#5

5. Let **S** be an re (recursively enumerable), non-recursive set, and **T** be an re, NON-EMPTY, possibly recursive set. Let

   E = { z | z = x + y, where x $\in$ S and y $\in$ T }.

   Answer with proofs, algorithms or counterexamples, as appropriate, each of the following questions:

   (a) Can **E** be non re?

   (b) Can **E** be re non-recursive?

   (c) Can **E** be recursive?

SEMI-THUE SYSTEMS

$S = (\Sigma, R)$

R: FINITE SET OF RULES

$\alpha \to \beta \quad \alpha, \beta \in \Sigma^*$

LIKE GRAMMARS BUT NO VARIABLES

CAN SIMULATE TMs (MAYBE DO LATER)

CAN SIMULATE STs BY POST CORRESPONDENCE PROBLEM. (PCP)

$\vec{X} = (x_1, \ldots, x_n), \vec{Y} = (y_1, \ldots, y_n), n > 0$

$x_i, y_i \in \Sigma^*$

$\exists\, i_1, \ldots, i_k, k > 0, 1 \le i_j \le n \ni$

$x_{i_1} \cdots x_{i_k} = y_{i_1} \cdots y_{i_k}$

$n = 3, \Sigma = \{a, b\}, \vec{X} = (aba, bb, a), \vec{Y} = (bab, b, baa)$

SOLUTION 2, 3, 1, 2 bbaabab b

bbaababb

IF ONE SOLN, $\exists$ AN INF. # OF SOLUTIONS

PROBLEM IS, IN GENERAL, UNDEC.

ABIGUITY OF CFG

$S \rightarrow A \mid B$

$\left.\begin{array}{l} A \rightarrow x_i A[i] \mid x_i [i] \\ B \rightarrow y_i B[i] \mid y_i [i] \end{array}\right\} 1 \le i \le n$

$\left.\begin{array}{l} A \stackrel{*}{\Rightarrow} x_{i_1} \cdots x_{i_k} [i_k] \cdots [i_1] \\ B \stackrel{*}{\Rightarrow} y_{i_1} \cdots y_{i_k} [i_k] \cdots [i_1] \end{array}\right\} k > 0$

AMBIGUOUS IFF SOLUTION TO INSTANCE OF PCP

INTERSECTION OF CFL

$L_A \cap L_B \neq \emptyset$ IFF SOLN OF PCP

THUS, EMPTINESS OF CSL UNDEC.

OUR EXAMPLE $\vec{X} = (aba, bb, a)$, $\vec{Y} = (bab, b, baa)$

$S \rightarrow A \mid B$

$A \rightarrow aba\,A[1] \mid bb\,A[2] \mid a\,A[3] \mid aba[1] \mid bb[2] \mid a[3]$

$B \rightarrow bab\,B[1] \mid b\,B[2] \mid baa\,B[3] \mid bab[1] \mid b[2] \mid baa[3]$

NON-EMPTINESS OF CSL

$S \to x_i S y_i^R \mid x_i T y_i^R \quad 1 \le i \le n$

$aTa \to *T*$

$*a \to a*$

$a* \to *a$

$T \to *$

$\Sigma = \{*\}$

$V = \{S, T\} \cup$ PCF ALPH.

GET STRINGS $*^{2j+1}$ for SOME j's

IFF SOLN TO PCP

ABOVE ALSO SHOWS FINITENESS (INFINITENESS) OF CSL IS UNDEC.

SEMI-THUE

$\alpha \to \beta$

AND TURING MACHINES
(SHORT STRING REPLACEMENT IN IDs)

CHANGES TO TM INSTANTANEOUS DESCRIPTION AS TM MAKES MOVES

ID IS SOME FINITE DESCRIPTION. USE ONE WITH LEFT/RIGHT SIDE STRINGS AND STATE

$\underbrace{\boxed{\cdots ? \mid a\,q\,b \mid ? \cdots}}_{\text{INCLUDES ALL NON-BLANK SQUARES}}$

$q\,b\,R\,p$

$\boxed{\cdots ?\,a\,b\,p\,? \cdots}$

$q\,b\,L\,p$

$\boxed{\cdots ?\,p\,a\,b\,? \cdots}$

$q\,b\,c\,p$

$\boxed{\cdots ?\,a\,p\,c\,? \cdots}$

TYPICALLY USE SOME LEFT/RIGHT END MARKERS

E.G. $\boxed{h \cdots \quad \cdots h}$

SO CAN EXPAND AND CONTRACT

FOR EXAMPLE $aqbh \Rightarrow abp0h$ IF RIGHT MOVE

OR $hqb \Rightarrow hq0b$ IF LEFT MOVE

MORE DETALS ON ST SIMULATING TM

$h \alpha q a \beta h$

IN STATE q, READING a
SEE PAGE 360 FOR DETAILED SIM

TM HALT PROBLEM CAN BE RECAST AS ST WORD PROBLEM

$$h 1^x q_1 0 h \xRightarrow{*} h \alpha q_0 \beta h \xRightarrow{*} h q_0 h$$

START

HALT STATE (USE JUST ONE)

CLEAN UP

THUS, IF TM, M, ACCEPT X, THAT IS, M HALTS WHEN STARTED ON X, THEN ST_M (ST SIMULATING M) REWRITES $h 1^x q_1 0 h$ AS $h q_0 h$

$$\left.\begin{matrix} h 1^x q_1 0 h \\ \text{AS } h q_0 h \end{matrix}\right\} \; h 1^x q_1 0 h \xRightarrow{*} h q_0 h$$

UNSOLVABILITY OF ST WORD PROBLEM

PRIOR SHOWS A SOLUTION TO ST WORD PROBLEM: GIVEN ST, S, AND WORDS w_1, w_2, IS IT TRUE THAT $w_1 \xRightarrow[S]{*} w_2$

IS UNDECIDABLE

UNSOLVABILITY OF PCP

CAN MIMIC ST DERIVATION BY SOLUTION TO INSTANCE OF PCP (POST CORRESPONDENCE PROB)

EXAMPLE: $aba \to ab$; $a \to aa$; $b \to a$ (ST, S)

QUESTION: $bbbb \xRightarrow[S]{*} aa$

PCP INSTANCE

$\vec{x}$	$[bbbb*$,	ab,	$\underline{ab}$,	aa,	$\underline{aa}$,	a,	$\underline{a}$,	$]$
$\vec{y}$	$[$,	$\underline{aba}$,	aba,	$\underline{a}$,	a,	$\underline{b}$,	b,	$\underline{*}aa]$

MUST START — RULES — MUST END

ALSO $*$, $\underline{*}$, a, $\underline{a}$, b, $\underline{b}$

$\underline{*}$, $*$, $\underline{a}$, a, $\underline{b}$, b

USED FOR PART NOT REWRITTEN

ST SOLUTION MAPPED TO PCP

MUST START [bbbb*
[

AS ONLY PAIR THAT STARTS WITH SAME CHARACTER

[bbbb*
[

MIMIC THREE STEPS OF b→a TO 1,2,4 CHARACTERS

[bbbb*abaa*
[bbbb*

MIMIC ONE STEP OF aba→ab

[bbbb*abaa*aba*
[bbbb*abaa*

MIMIC ONE STEP OF aba→ab

[bbbb*abaa*aba*ab*
[bbbb*abaa*aba*
[bbbb*abaa*

MIMIC ONE STEP OF b→a

[bbbb*abaa*aba*ab*aa
[bbbb*abaa*aba*ab

[bbbb*abaa*aba*ab*aa]
[bbbb*abaa*aba*ab*aa]
[bbbb*abaa*

LAST STEP

PCP CONSEQUENCES

P: $\vec{X} = (x_1, \ldots, x_n)$ $\vec{Y} = (y_1, \ldots, y_n)$

① AMBIGUITY OF CFG IS UNDECIDABLE

$S \to A \mid B$

$A \to x_i A [i] \mid x_i [i]$ $1 \le i \le n$

$B \to y_i B [i] \mid y_i [i]$ $1 \le i \le n$

P HAS SOLN. IFF $\exists i_1 \ldots i_k \ni$ $\begin{cases} A \overset{*}{\Rightarrow} x_{i_1} \cdots x_{i_k} [i_k] \cdots [i_1] & k > 0 \\ B \overset{*}{\Rightarrow} y_{i_1} \cdots y_{i_k} [i_k] \cdots [i_1] \\ x_{i_1} \cdots x_{i_k} = y_{i_1} \cdots y_{i_k} \end{cases}$

IFF ABOVE GRAMMAR IS AMBIGUOUS

② CONSIDER A-RULES AND B-RULES ABOVE

$L_A \cap L_B \ne \phi$ IFF P HAS SOLN

SO

NON-EMPTINESS (EMPTINESS) OF INTERSECTION OF CFLs IS UNDEC.

NOTE! INTERSECTION OF CFLs IS A CSL SO EMPTINESS OF CSL IS UNDEC.

③ CAN DO CSL EMPTINESS AS DIRECT PROOF

$S \to x_i S y_i^R \mid x_i T y_i^R$ $1 \le i \le n$

$aTa \to *T*$

$*a \to a*$

$a* \to *a$

$T \to *$

TRACES

A COMPUTATIONAL TRACE IS A WORD OF FORM

$\#X_0\#X_1\# \ldots \#X_k\#$

WHERE $X_i \Rightarrow X_{i+1}$ $0 \le i < k$, EACH X_i AN ID, AND X_k IS A TERMINAL ID

WE CAN SHOW CFGs THAT ALMOST GET TRACES (OFTEN ALTERNATING BETWEEN IDs AND THE REVERSAL OF IDs), WHAT THEY GET IS EVERY OTHER PAIR RIGHT

$\#X_0\#X_1\# \ldots \#X_k\#$

WHERE $X_{2i} \Rightarrow X_{2i+1}$

BUT $X_{2i+1} \Rightarrow ? X_{2i+2}$

SO EVEN/ODD RIGHT, BUT ODD/EVEN ARE NOT NEC. RIGHT

CAN GET SEPARATE GRAMMAR FOR ODD/EVEN

CAN USE $\cap$ OR QUOTIENT TO GET ALIGNMENT

WITHOUT PROOF (SEE 376), FOR CFL L_1, L_2

L_1/L_2 CAN BE ANY RE SET.

COMPLEMENT OF TRACE

WHILE TRACE REQUIRES ALL PAIRS TO BE RIGHT, COMPLEMENT REQUIRES JUST ONE ERROR.

CAN SHOW COMPLEMENTS OF TRACES ARE CFLs. MOREOVER, IF MACHINE FOR WHICH WE HAVE TRACES ACCEPTS $\emptyset$ THEN THERE ARE NO TERMINATING TRACES AND COMPLEMENT IS Σ^*.

THIS IS WHY WE CANNOT DECIDE IF A CFG, G, IS SUCH THAT $\mathcal{L}(G)=\Sigma^*$

WE CAN DECIDE IF $\mathcal{L}(G)=\emptyset$, $|\mathcal{L}(G)|=$ INF.

$$\mathcal{L}(G) = \mathcal{L}(G)^2$$

LET G BE AN ARBITRARY CFG (OR CSG)
LET $L = \mathcal{L}(G)$. WANT TO SHOW $L = L^2$ IS UNDEC.
NOTE: $L^2 = \{xy \mid x, y \in L\}$
FIRST, RECALL ~~THAT~~ $L = \Sigma^*$ IS UNDEC.
FOR CFG (AND CSG)

CLAIM: $L = \Sigma^*$ IFF
(1) $\Sigma \cup \{\lambda\} \subseteq L$; AND (NOTE: DECIDABLE)
(2) $L \cdot L = L$
CLEARLY IF $L = \Sigma^*$ THEN (1) AND (2) HOLD
CONVERSELY, $\Sigma^* \subseteq L^* = \bigcup_{n \geq 0} L^n \subseteq L$

FOLLOWS FROM (1)

FOLLOWS FROM (2) SINCE IF $L = L^2$ THEN $L = L^3$, ETC.